WHAT IS CLAIMED IS:

1. A method for providing user interfaces for a plurality of services offered by an information distribution system, comprising:

providing a first application to support a first user interface for a first service;

providing a second application to support a second user interface for a second service; and

coordinating passing of control between the first and second applications via a control mechanism.

2. The method of claim 1, further comprising:

maintaining first and second message queues for the first and second applications, respectively.

3. The method of claim 2, further comprising:

passing control to the first and second applications via messages provided to the first and second message queues, respectively.

4. The method of claim 1, further comprising:

polling the first or second application to determine a status of the application.

5. The method of claim 2, further comprising:

polling for a status of the first or second application by providing a poll message to the first or second message queue, respectively.

6. The method of claim 1, further comprising:

providing a root application to support communication between the first and second applications and a lower layer.

7. The method of claim 6, wherein the communication between the root application and the first and second applications is achieved via a set of application programming interfaces (APIs).

8. The method of claim 6, wherein the lower layer is a hardware layer.

9. The method of claim 1, wherein each of the first and second applications is operable in an active state or an inactive state.

10. The method of claim 9, wherein an active application is operative to process key inputs.

11. The method of claim 9, wherein the first application transitions to the inactive state upon occurrence of any one of a plurality of events in a first set, and the second application transitions to the inactive state upon occurrence of any one of a plurality of events in a second set.

12. The method of claim 11, wherein the plurality of events in the first set includes a first set of key presses, and the plurality of events in the second set includes a second set of key presses.

13. The method of claim 9, wherein the first and second applications transition to the active state in response to receiving a launch message in the first and second message queues, respectively.

14. The method of claim 9, wherein the first and second applications transition to the active state in response to receiving first and second key presses, respectively.

15. The method of claim 1, further comprising:

providing a first link in the first user interface to activate the second user interface; and

providing a second link in the second user interface to activate the first user interface.

16. The method of claim 1, wherein only the first or second application, if any, is active at any particular moment.

17. The method of claim 1, wherein each of the first and second applications is independently executed.

18. The method of claim 1, wherein the first and second applications are concurrently active or semi-active.

19. The method of claim 1, wherein the first application supports an interactive program guide (IPG).

20. The method of claim 19, wherein the second application supports video-on-demand (VOD).

21. The method of claim 20, wherein the first application is operable to overlay at least a portion of a VOD user interface on top of an IPG user interface.

22. The method of claim 20, wherein the second application is operable to overlay at least a portion of an IPG user interface on top of a VOD user interface.

23. The method of claim 20, wherein the first and second applications are operable to overlay a channel information window on top of an IPG user interface and a VOD user interface, respectively.

24. A method for providing interactive program guide (IPG) and video-on-demand (VOD) user interfaces for IPG and VOD services, comprising:

providing an IPG application to support the IPG user interface for the IPG service;

providing a VOD application to support the VOD user interface for the VOD service;

maintaining IPG and VOD message queues for the IPG and VOD applications, respectively; and

passing control to the IPG and VOD applications via messages provided to the IPG and VOD message queues, respectively.

25. A terminal configurable to provide user interfaces for a plurality of services offered by an information distribution system, comprising:

a first application operable to support a first user interface for a first service;

a second application operable to support a second user interface for a second service; and

means for passing control between the first and second applications.

26. The terminal of claim 25, further comprising:

a root application operable to support communication between the first and second applications and a hardware layer.

27. The terminal of claim 25, further comprising:

first and second message queues operable to store messages for the first and second applications, respectively.

28. The terminal of claim 27, wherein the means for passing control is implemented by providing messages to the first and second message queues, and wherein the first and second applications are operable to retrieve and process messages stored in the first and second message queues, respectively.

29. The terminal of claim 25, wherein the first application supports an interactive program guide (IPG) and the second application supports video-on-demand (VOD).

30. A terminal configurable to provide user interfaces for a plurality of services offered by an information distribution system, comprising:

a first state indicative of a first application executing to support a first user interface for a first service;

a second state indicative of a second application executing to support a second user interface for a second service;

a third state indicative of the first and second applications being idle; and

means for transitioning between the first, second, and third states.

31. The terminal of claim 25, wherein transitions between the first, second, and third states are in response to defined key presses.